**Summary of Basic Commands**

| **Action** | **Files** | **Folders** |
| --- | --- | --- |
| Inspect | ls | ls |
| View content | cat | ls |
| Navigate to |  | cd |
| Move | mv | mv |
| Copy | cp | cp -r |
| Create | nano | mkdir |
| Delete | rm | rmdir, rm -r |

**Filesystem hierarchy**

The following is an overview of a standard Unix filesystem. The exact hierarchy depends on the platform. Your file/directory structure may differ slightly:

**Glossary**

**absolute path**

A [path](https://swcarpentry.github.io/shell-novice/reference.html#path) that refers to a particular location in a file system. Absolute paths are usually written with respect to the file system’s [root directory](https://swcarpentry.github.io/shell-novice/reference.html#root-directory), and begin with either “/” (on Unix) or “\” (on Microsoft Windows). See also: [relative path](https://swcarpentry.github.io/shell-novice/reference.html#relative-path).

**argument**

A value given to a function or program when it runs. The term is often used interchangeably (and inconsistently) with [parameter](https://swcarpentry.github.io/shell-novice/reference.html#parameter).

**command shell**

See [shell](https://swcarpentry.github.io/shell-novice/reference.html#shell)

**command-line interface**

A user interface based on typing commands, usually at a [REPL](https://swcarpentry.github.io/shell-novice/reference.html#read-evaluate-print-loop). See also: [graphical user interface](https://swcarpentry.github.io/shell-novice/reference.html#graphical-user-interface).

**comment**

A remark in a program that is intended to help human readers understand what is going on, but is ignored by the computer. Comments in Python, R, and the Unix shell start with a # character and run to the end of the line; comments in SQL start with --, and other languages have other conventions.

**current working directory**

The directory that [relative paths](https://swcarpentry.github.io/shell-novice/reference.html#relative-path) are calculated from; equivalently, the place where files referenced by name only are searched for. Every [process](https://swcarpentry.github.io/shell-novice/reference.html#process) has a current working directory. The current working directory is usually referred to using the shorthand notation . (pronounced “dot”).

**file system**

A set of files, directories, and I/O devices (such as keyboards and screens). A file system may be spread across many physical devices, or many file systems may be stored on a single physical device; the [operating system](https://swcarpentry.github.io/shell-novice/reference.html#operating-system) manages access.

**filename extension**

The portion of a file’s name that comes after the final “.” character. By convention this identifies the file’s type: .txt means “text file”, .png means “Portable Network Graphics file”, and so on. These conventions are not enforced by most operating systems: it is perfectly possible (but confusing!) to name an MP3 sound file homepage.html. Since many applications use filename extensions to identify the [MIME type](https://swcarpentry.github.io/shell-novice/reference.html#mime-type) of the file, misnaming files may cause those applications to fail.

**filter**

A program that transforms a stream of data. Many Unix command-line tools are written as filters: they read data from [standard input](https://swcarpentry.github.io/shell-novice/reference.html#standard-input), process it, and write the result to [standard output](https://swcarpentry.github.io/shell-novice/reference.html#standard-output).

**for loop**

A loop that is executed once for each value in some kind of set, list, or range. See also: [while loop](https://swcarpentry.github.io/shell-novice/reference.html#while-loop).

**graphical user interface**

A user interface based on selecting items and actions from a graphical display, usually controlled by using a mouse. See also: [command-line interface](https://swcarpentry.github.io/shell-novice/reference.html#command-line-interface).

**home directory**

The default directory associated with an account on a computer system. By convention, all of a user’s files are stored in or below her home directory.

**loop**

A set of instructions to be executed multiple times. Consists of a [loop body](https://swcarpentry.github.io/shell-novice/reference.html#loop-body) and (usually) a condition for exiting the loop. See also [for loop](https://swcarpentry.github.io/shell-novice/reference.html#for-loop) and [while loop](https://swcarpentry.github.io/shell-novice/reference.html#while-loop).

**loop body**

The set of statements or commands that are repeated inside a [for loop](https://swcarpentry.github.io/shell-novice/reference.html#for-loop) or [while loop](https://swcarpentry.github.io/shell-novice/reference.html#while-loop).

**MIME type**

MIME (Multi-Purpose Internet Mail Extensions) types describe different file types for exchange on the Internet, for example, images, audio, and documents.

**operating system**

Software that manages interactions between users, hardware, and software [processes](https://swcarpentry.github.io/shell-novice/reference.html#process). Common examples are Linux, macOS, and Windows.

**option**

A way to specify an argument or setting to a command-line program. By convention Unix applications use a dash followed by a single letter, such as -v, or two dashes followed by a word, such as --verbose, while DOS applications use a slash, such as /V. Depending on the application, an option may be followed by a single argument, as in -o /tmp/output.txt.

**parameter**

A variable named in a function’s declaration that is used to hold a value passed into the call. The term is often used interchangeably (and inconsistently) with [argument](https://swcarpentry.github.io/shell-novice/reference.html#argument).

**parent directory**

The directory that “contains” the one in question. Every directory in a file system except the [root directory](https://swcarpentry.github.io/shell-novice/reference.html#root-directory) has a parent. A directory’s parent is usually referred to using the shorthand notation .. (pronounced “dot dot”).

**path**

A description that specifies the location of a file or directory within a [file system](https://swcarpentry.github.io/shell-novice/reference.html#file-system). See also: [absolute path](https://swcarpentry.github.io/shell-novice/reference.html#absolute-path), [relative path](https://swcarpentry.github.io/shell-novice/reference.html#relative-path).

**pipe**

A connection from the output of one program to the input of another. When two or more programs are connected in this way, they are called a “pipeline”.

**process**

A running instance of a program, containing code, variable values, open files and network connections, and so on. Processes are the “actors” that the [operating system](https://swcarpentry.github.io/shell-novice/reference.html#operating-system) manages; it typically runs each process for a few milliseconds at a time to give the impression that they are executing simultaneously.

**prompt**

A character or characters display by a [REPL](https://swcarpentry.github.io/shell-novice/reference.html#read-evaluate-print-loop) to show that it is waiting for its next command.

**quoting**

(in the shell): Using quotation marks of various kinds to prevent the shell from interpreting special characters. For example, to pass the string \*.txt to a program, it is usually necessary to write it as '\*.txt' (with single quotes) so that the shell will not try to expand the \* wildcard.

**read-evaluate-print loop**

(REPL): A [command-line interface](https://swcarpentry.github.io/shell-novice/reference.html#command-line-interface) that reads a command from the user, executes it, prints the result, and waits for another command.

**redirect**

To send a command’s output to a file rather than to the screen or another command, or equivalently to read a command’s input from a file.

**regular expression**

A pattern that specifies a set of character strings. REs are most often used to find sequences of characters in strings.

**relative path**

A [path](https://swcarpentry.github.io/shell-novice/reference.html#path) that specifies the location of a file or directory with respect to the [current working directory](https://swcarpentry.github.io/shell-novice/reference.html#current-working-directory). Any path that does not begin with a separator character (“/” or “\”) is a relative path. See also: [absolute path](https://swcarpentry.github.io/shell-novice/reference.html#absolute-path).

**root directory**

The top-most directory in a [file system](https://swcarpentry.github.io/shell-novice/reference.html#file-system). Its name is “/” on Unix (including Linux and macOS) and “\” on Microsoft Windows.

**shell**

A [command-line interface](https://swcarpentry.github.io/shell-novice/reference.html#command-line-interface) such as Bash (the Bourne-Again Shell) or the Microsoft Windows DOS shell that allows a user to interact with the [operating system](https://swcarpentry.github.io/shell-novice/reference.html#operating-system).

**shell script**

A set of [shell](https://swcarpentry.github.io/shell-novice/reference.html#shell) commands stored in a file for re-use. A shell script is a program executed by the shell; the name “script” is used for historical reasons.

**standard input**

A process’s default input stream. In interactive command-line applications, it is typically connected to the keyboard; in a [pipe](https://swcarpentry.github.io/shell-novice/reference.html#pipe), it receives data from the [standard output](https://swcarpentry.github.io/shell-novice/reference.html#standard-output) of the preceding process.

**standard output**

A process’s default output stream. In interactive command-line applications, data sent to standard output is displayed on the screen; in a [pipe](https://swcarpentry.github.io/shell-novice/reference.html#pipe), it is passed to the [standard input](https://swcarpentry.github.io/shell-novice/reference.html#standard-input) of the next process.

**sub-directory**

A directory contained within another directory.

**tab completion**

A feature provided by many interactive systems in which pressing the Tab key triggers automatic completion of the current word or command.

**variable**

A name in a program that is associated with a value or a collection of values.

**while loop**

A loop that keeps executing as long as some condition is true. See also: [for loop](https://swcarpentry.github.io/shell-novice/reference.html#for-loop).

**wildcard**

A character used in pattern matching. In the Unix shell, the wildcard \* matches zero or more characters, so that \*.txt matches all files whose names end in .txt.

**External references**